UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/028,858 | 12/19/2001 | Shivnandan D. Kaushik | 042390.P13163 | 4672 |

45209 7590 03/30/2009

INTEL/BSTZ
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP
1279 OAKMEAD PARKWAY
SUNNYVALE, CA 94085-4040

| EXAMINER | |
|---|---|
| ZAMAN, FAISAL M | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2111 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 03/30/2009 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

---

*Ex parte* SHIVNANDAN D. KAUSHIK,
LING CEN, JAMES B. CROSSLAND,
MOHAN J. KUMAR, LINDA J. RANKIN,
and DAVID J. O'SHEA

---

Appeal 2008-2924
Application 10/028,858
Technology Center 2100

---

Decided:[1] March 30, 2009

---

Before LEE E. BARRETT, ST. JOHN COURTENAY, III, and
THU A. DANG, *Administrative Patent Judges*.

BARRETT, *Administrative Patent Judge*.

DECISION ON APPEAL

This is a decision on appeal under 35 U.S.C. § 134(a) from the final rejection of claims 1-5, 7-10, 28, 29, 35, and 37. Claims 16-27 and 31-34

---

[1] The two-month time period for filing an appeal or commencing a civil action, as recited in 37 C.F.R. § 1.304, begins to run from the decided date shown on this page of the decision. The time period does not run from the Mail Date (paper delivery) or Notification Date (electronic delivery).

have been allowed. Claims 11-15 have been canceled. Claims 6, 30, 36, and 38 have been objected to and have been indicated to be allowable if rewritten in independent form. We have jurisdiction under 35 U.S.C. § 6(b).

We affirm-in-part.

## STATEMENT OF THE CASE

### *Appellants' invention*

Hot plugging is a way to add memory and other kinds of adapter cards to a running computer without rebooting. The invention relates to methods for adding hot plug modules to a running computer device including the step of enabling a communication interface on the module to establish a communication link with the running computer device (independent claims 1 and 8) and a computing device having a midplane[2] wherein the state of the hot plug interface is tracked to indicate when resources are ready to join the computing device (independent claim 24).

[2] A "midplane" is a circuit board where cards and devices can be connected to both sides.

***Claims***

Representative claims 1 and 28 are reproduced below:

1. A method of adding one or more caching agents to a running computer device, comprising

identifying the one or more caching agents provided by a hot plug module in response to the hot plug module being physically coupled to the running computer device;

adding the identified caching agents of the hot plug module to a resource pool of the running computer device; and

enabling a communication interface on the hot plug module to establish a communication link with the running computer device.

28. A computing device comprising,

a midplane comprising a coupler and a hot plug interface to track a state associated with the coupler;

a hot plug module comprising a coupler to detachably couple the hot plug module to the coupler of the midplane and resources coupled to the coupler of the hot plug module via a hot plug interface of the hot plug module, the hot plug module to update the state of the hot plug interface of the midplane to indicate when the resources are ready to join the computing device, and

a processor coupled to the hot plug interface of the midplane, the processor to add the resources to the computing device without rebooting in response to determining that the hot plug interface of the midplane indicates the resources are ready to join.

***The references***[3]

| | | |
|---|---|---|
| Bealkowski | US 6,282,596 B1 | Aug. 28, 2001 |
| Olarig | US 6,587,909 B1 | Jul. 01, 2003 (filed Mar. 6, 2000) |

***The rejections***

Claims 1 and 7-9 stand rejected under 35 U.S.C. § 102(e) as anticipated by Olarig.

Claims 1-5, 7-10, 28, 29, 35, and 37 stand rejected under 35 U.S.C. § 102(e) as anticipated by Bealkowski.

DISCUSSION

***Claim interpretation***

Before we can address the rejections, we must first interpret the limitation "enabling a communication interface on the hot plug module to establish a communication link with the running computer device," as recited in independent claims 1 and 8. Appellants refer to blocks 1001-1002 of Fig. 9H and paragraphs [0084]-[0085] for a description of the limitation (Br. 3). Paragraphs [0084]-[0085] state:

> [0084] In response to passing the self test, the hot plug module 110 may enable in block 1001 its communication interfaces 214 to establish a communication link with communication interfaces 628 of the switches 624 and to couple signals of the hot plug module 110 to

[3] The Examiner cites *PCI Local Bus Specification*, Revision 2.2, PCI Special Interest Group, December 18, 1998, pages 26-32 to show that PCI buses transmit data using packets (Ans. 15). However, the reference is not applied in a ground of rejection.

> signals of the running computing device 100. For example, a processor 230 of the hot plug module 110 or a system management processor 636 of the midplane 120 may enable the communication interfaces 214 by setting associated interface enable fields 746 of the hot plug module 110.
>
> [0085] In block 1002, the enabled communication interfaces 214 of the hot plug module 110 may establish a communication link with the communication interfaces 628 of the midplane 120. The communication interfaces 214, 628 may initiate a framing process in response to being enabled. As a result of the framing process, the interface 5 logic 704 associated with the communication interfaces 214, 628 may set the framing detected fields 756 in response to detecting a predetermined number of framing packets and may set the acknowledgement detected fields 758 in response to detecting a predetermined number of acknowledgement packets. It should be appreciated that the default switch 624 may generate a hot plug interrupt in response to the interface logic detecting a predetermined number of framing packets and setting the framing detected field 756 of the default switch 624.

We further refer to Figure 7 which shows the hot plug interface associated with a communication interface of the computing device of FIG. 1, where "[i]t should be appreciated that the hot plug interfaces 212 of the hot plug modules 110 may be implemented in a similar manner" ([0038]). As shown in Figure 7, the communication interface 628 is enabled by a setting a bit in the enable interface field 746. That is, "a processor 230 of the hot plug module 110 . . . may enable the communication interfaces 214 by setting associated interface enable fields 746 of the hot plug module 110" ([0084]).

An "interface" is defined as a "shared electrical boundary between parts of a computer system, through which information is conveyed." *The*

*New IEEE Standard Dictionary of Electrical and Electronics Terms* (IEEE, Inc., 5th ed. 1993). Thus, although Appellants describe a block that is controlled by an enable signal in Figure 7, we interpret "communication interface on the hot plug module" broadly to be the module side of the boundary between the hot plug module and the computer device through which communications take place. The boundary is defined by the connector between the computer and the hot plug module. However, since the communication interface must "establish a communication link with the running computer" when enabled, the communication interface must also include whatever circuitry is necessary to communicate through the interface between the module and the computer (i.e., through the connector). This circuitry is enabled when power, clock signals, data, address, and control signals are applied from the computer system to the hot plug module.

The limitation of "enabling a communication interface on the hot plug module to establish a communication link with the running computer device" does not require "enabling" by setting a bit. The term "enable" is defined as a "command or condition which permits some specific event to proceed," *The New IEEE Standard Dictionary*. We interpret "enabling a communication interface on the hot plug module to establish a communication link with the running computer device" to include any command or condition which allows the hot plug interface module to communicate with the running computer device. The Examiner states that "the term 'enabling' is a broad term, which can be taken to mean simply to allow a function to be performed" (Ans. 11), which is consistent with our

definition. Also, the limitation does not define where the "enabling" of the communication interface takes place: the "communication interface on the hot plug module" is "enabled" by turning on power, clock, address, data, and control lines to the hot plug module connector at the computer side.

If Appellants want to limit the claims to a communication interface on the hot plug module having some control input to enable the interface, this can be done by amendment. *See In re Zletz*, 893 F.2d 319, 321-22 (Fed. Cir. 1989) ("[D]uring patent prosecution when claims can be amended, ambiguities should be recognized, scope and breadth of language explored, and clarification imposed. . . . An essential purpose of patent examination is to fashion claims that are precise, clear, correct, and unambiguous. Only in this way can uncertainties of claim scope be removed, as much as possible, during the administrative process.").

***Olarig***

*Claims 1 and 7-9*

*Contentions*

Appellants argue that Olarig does not teach "enabling a communication interface on the hot plug module to establish a communication link with the running computer device," as recited in independent claims 1 and 8.

The Examiner finds that the five tasks described at column 5, lines 1-23, enable a communication interface to establish a communication link (Final Rej. 5).

Appellants argue that the Examiner appears to equate turning on the power before driving the circuitry with a clock signal in column 6, lines 50-57, with "enabling a communication interface on the hot plug module," and that one of ordinary skill in the art would not equate turning on the power with enabling the communication interface (Br. 8).

The Examiner states:

> Contrary to Appellant's argument, in the cited section (Column 6, lines 50-57) the newly added memory module is not enabled to communicate with the computer system (ie. transfer data to/from other components in the system) until the circuitry on the memory module is fully turned on. Further, the term "enabling" is a broad term, which can be taken to mean simply to allow a function to be performed. In the context of the Olarig reference, the "enabling" function can be equated to the computer system 10 sending a signal to activate power within memory modules 14 as they are inserted into slots 16.

Ans. 11-12.

Appellants argue that Olarig describes a power up sequence for memory modules, but "data transfer can not take place unless a communication interface meant for the transfer of data is enabled" (Reply Br. 4). It is argued that a person skilled in the art would not equate turning on the power to be similar to "enabling a communication interface on the hot plug module to establish a communication link" (*id.*).

Appellants argued in the Reply Brief that the Examiner has presumed that slot connector 16 in Olarig has a communication interface. "Applicants have respectfully and repeatedly submitted that the slot connector is used to provide power to turn on the circuitry on the memory module and does not

appear to have a communication interface. Appellants submit that the slot connector should have a built-in intelligence to establish the communication link." Reply Br. 3.

*Issue*

Does Olarig teach "enabling a communication interface on the hot plug module to establish a communication link with the running computer device," as recited in claims 1 and 8?

*Analysis*

Initially, we do not find Appellants' argument that Olarig does not have a communication interface in the main Brief. Arguments presented for the first time in the Reply Brief are untimely. *Cf. Kaufman Company, Inc. v. Lantech, Inc.*, 807 F.2d 970, 973 n.* (Fed. Cir. 1986); *McBride v. Merrell Dow and Pharmaceuticals*, Inc., 800 F.2d 1208, 1210-11 (D.C. Cir. 1986) ("We generally will not entertain arguments omitted from an appellant's opening brief and raised initially in his reply brief. . . . Considering an argument advanced for the first time in a reply brief, then, is not only unfair to an appellee, . . . but also entails the risk of an improvident or ill-advised opinion on the legal issues tendered."). Nevertheless, as discussed in the claim interpretation there is communication going on between the computer and the module in Olarig and the module has an interface to the computer system at the connector; thus, Olarig has a communication interface.

As discussed in the claim interpretation section, we interpret "enabling a communication interface on the hot plug module to establish a

communication link with the running computer device" to met by any command or condition which allows the hot plug interface module to communicate with the running computer device. While providing power and a clock line to the hot plug module as described at column 6, lines 50-57, of Olarig are necessary to establishing communication, power and clock are not sufficient to "establish a communication link." Similarly, configuring the memory at column 5, lines 1-23, does not "establish a communication link." A better teaching in Olarig is the following:

> In controlling the connection/disconnection of a particular slot connector 16, the memory controller 16 regulates when power is connected to the connector 16 and also regulates when other signals are connected to the slot connector 16. For example, in some arrangements, the memory controller 12 selectively regulates when a system clock line, address lines, data lines and control lines are connected to the slot connectors 16.

Olarig, col. 6, ll. 28-35; *see also* col. 6, ll. 64-65 ("Next, the address, data and control lines of the memory bus 34 are connected to the slot connector 16."). Connecting the address, data, and control lines to the module establishes communication between the module and the computer.

Olarig teaches "enabling a communication interface on the hot plug module to establish a communication link with the running computer device," as recited in independent claims 1 and 8. The rejection of claims 1 and 8 is affirmed. Dependent claims 7 and 9 are not separately argued, so the rejection of these claims is affirmed.

***Bealkowski***

*Claims 1, 5, 8, and 10*

*Contentions*

Appellants argue that Bealkowski does not teach "enabling a communication interface on the hot plug module to establish a communication link with the running computer device," as recited in independent claims 1 and 8.

Appellants argue that Bealkowski discloses a hot-pluggable system bus 18 for processor cards 11a-11d, but "the processor cards 11a-11d appear not to have a communication interface" (Br. 10) and, "[t]hus, Bealkowski appears to provide no teaching regarding enabling a communication interface of the processor cards 11a-11d" (*id.*). It is argued that Bealkowski discloses FET switches 82, 86 to control power to the processor cards 11a-11d, a clock buffer to control application of clock signals, and FET switches 80 that provide front-side isolation to maintain electrical integrity during hot-plug. "However, . . . the FET switches 80, 82, 86 and the clock buffer are all on the computer system side of the CPU connectors 14 and are not part of the processor cards 11a-11d." *Id.* It is argued that even if the processor cards inherently have an interface for mating with the CPU connectors 14, Bealkowski does not inherently teach that such an interface is enabled (Br. 10-11).

The Examiner finds that since the processor cards 11a-11d communicate on the bus 18, they must have a communication interface to connect to connectors 14a-14d (Ans. 12). The Examiner finds that

Bealkowski teaches supplying power along with a clock signal to processor cards 11a-11d once a processor card is coupled to the system bus 18, "which is equivalent to enabling the communication interface on the processor card" (*id.*), where "'enabling' can be taken to mean simply to allow a function to be performed" (*id.*).

Appellants, in addition to repeating arguments in the Brief, argue that Bealkowski teaches that supplying power and transmitting initialization data are independent of the processor bus and therefore Bealkowski does not teach enabling a communication interface (Reply Br. 6-7).

*Issue*

Does Bealkowski teach "enabling a communication interface on the hot plug module to establish a communication link with the running computer device," as recited in independent claims 1 and 8?

*Analysis*

Initially, we agree with the Examiner that Bealkowski inherently has a communication interface because there is communication going on between the computer and the module. The communication interface starts at the module side of the connector and includes the circuitry necessary for performing the communications. As discussed in the claim interpretation section, "enabling a communication interface on the hot plug module to establish a communication link with the running computer device" does not require a separate control input and is met by any command or condition which allows the hot plug interface module to communicate with the running

computer device. "Enabling" the module can be performed on the computer side of the computer/module interface. While providing power and a clock line to the hot plug module is necessary for establishing communication, power and a clock signal do not "establish a communication link." The communication link is communication between the module and the running computer. Bealkowski discloses that after an applying power and clock signals, and conducting an initialization procedure, the FET switch 80 is enabled whereby all processors are placed into service during the same cycle (col. 7, ll. 22-23; Figures 4A and 4B), which we find is enabling the communication interface to establish a communication link.

Bealkowski teaches "enabling a communication interface on the hot plug module to establish a communication link with the running computer device," as recited in independent claims 1 and 8. The rejection of claims 1 and 8 is affirmed. Appellants do not argue the separate patentability of claims 5 and 10, so the rejection of claims 5 and 10 is affirmed.

*Claims 2-4 and 9*

*Contentions*

The Examiner finds that Bealkowski teaches enabling a communication interface, CPU connector 14, to establish a communication link (Final Rej. 8; Ans. 8)

Appellants argue that claims 2 and 9, and therefore claims 2-4, 9, and 10 which depend therefrom, require *both* a communication interface *on the hot plug module* and a communication interface *of a running computer*

*system* to establish the communication and this is not taught by Bealkowski (Br. 11-12; Reply Br. 9). Appellants argue that the Examiner appears to equate CPU connector 14 of Bealkowski with Appellants' communication interface on a hot plug module, but "submit that a person skilled in the art would not treat a CPU connector 14 as being similar to the communication interface of the hot plug module" (Br. 12; similar argument Reply Br. 7).

*Issue*

Does Bealkowski teach "enabling a communication interface of the running computing device that is associated with the hot plug module in response to determining that the hot plug module has been physically coupled to the running computing device," as recited in claims 2 and 9?

*Analysis*

After determining that a hot plug module has been physically attached to the running computer device, step 102 in Figure 4A, the procedure in Figures 4A and 4B of Bealkowski takes a series of steps including self-test and initialization before enabling the FET switch 80 to place all processors into service during the same cycle. Enabling the FET switch 80 establishes communication between the module and the computer and, therefore, enables a communication interface on both the module and the computer. The term "enabling a communication interface" is broad and does not imply that the interface has a special control input or where enabling takes place.

Bealkowski teaches "enabling a communication interface of the running computing device that is associated with the hot plug module in

response to determining that the hot plug module has been physically coupled to the running computing device," as recited in claims 2 and 9. The rejection of claims 2 and 9 is affirmed. Claims 3 and 4 have not been separately argued, so the rejection of these claims is affirmed.

*Claim 7*

*Contentions*

The Examiner finds that Bealkowski teaches adding the identified memory of the hot plug module to a memory pool of the running computing device at column 9, lines 42-45 (Final Rej. 9).

Appellants argue that column 9 only teaches that the caches are flushed and the processor is set to idle and does not teach all the limitations of claim 7 (Br. 13).

The Examiner states that column 9 indicates that caches of the processor cards 11a-11d need to be flushed when the cards are removed, which indicates that they were in the memory pool before it was taken out of the system (Ans. 14).

Appellants argue that Bealkowski does not teach adding identified memory to the memory pool to increase the memory pool or identifying memory before adding it to the memory pool (Reply Br. 8-9).

*Issue*

Does Bealkowski teach "identifying memory of a hot plug module . . . ; and adding the identified memory of the hot plug module to a memory

pool of the running computing device to increase the memory pool from which memory is allocated to processes"?

*Analysis*

Bealkowski describes hot plugging a data processor where each processor has at least one level of cache (col. 4, ll. 3-6). There is no teaching that the memory caches are identified and then pooled in a memory pool so that the computing device allocates memory from a memory pool. Each processor could be using its own cache to run its own processes. Absent some clearer teaching, it would be speculation to assume that Bealkowski operates as recited in claim 7.

Bealkowski does not teach "identifying memory of a hot plug module . . . ; and adding the identified memory of the hot plug module to a memory pool of the running computing device to increase the memory pool from which memory is allocated to processes." The rejection of claim 7 is reversed.

*Claims 35 and 37*

*Contentions*

The Examiner finds that Bealkowski teaches transferring packets at Figure 4 and column 7, line 51 to column 8, line 65 (Final Rej. 10).

Appellants argue that they are unable to locate the limitation, "transferring packets between the communication interface on the hot plug module and the communication interface of the running computer system to establish the communication link," in Bealkowski (Br. 15).

The Examiner further refers to column 4, lines 38-46, where the data must inherently be in the form of packets as evidenced by the PCI specification (Ans. 15).

Appellants argue that processors 30a-30d and memory element 32 are coupled to the PCI bus via a PCI Host Bridge 34, but that the PCI Host Bridge is not provided with the processors 30a-30d. Appellants argue that a "person skilled in the art would not treat PCI Host Bridge as being similar to the communication interface on the hot plug module and the communication interface of the running computer system" (Reply Br. 11).

*Issue*

Does Bealkowski teach "transferring packets between the communication interface on the hot plug module and the communication interface of the running computer system to establish the communication link"?

*Analysis*

The Examiner interprets the communication interface to be the connector 14a in Figure 3A, which is connected from processors 30a-30d to the system bus 18 in Figure 2. Assuming that the PCI Host Bridge transfers packets from the system bus 18 to the PCI Bus 37, there is no express teaching that data is transferred from the processors 30a-30d to the system bus using packets. While it may be logical that both buses would use packets, this is not a certainty.

Bealkowski does not teach "transferring packets between the communication interface on the hot plug module and the communication interface of the running computer system to establish the communication link." The rejection of claims 35 and 37 is reversed.

*Claim 28*

*Contentions*

Appellants argue that Bealkowski does not teach "a midplane comprising a coupler and a hot plug interface to track a state associated with the coupler; . . . the hot plug module to update the state of the hot plug interface of the midplane to indicate when the resources are ready to join the computing device; and a processor coupled to the hot plug interface of the midplane, the processor to add the resources to the computing device without rebooting in response to determining that the hot plug interface of the midplane indicates the resources are ready to join."

The Examiner refers to column 5, lines 41-52 for the limitation "to update the state of the hot plug interface of the midplane to indicate when the resources are ready to join the computing device."

Appellants argue that this portion of Bealkowski only discloses the control functions of the hot plug controller 70 and LED status indicators, and does not teach a hot plug module that updates a hot plug interface (Br. 14). Appellants argue that Bealkowski merely indicates that a service processor manages FET switches that isolate a processor card during addition and

removal and there is "no teaching of the processor updating a hot plug interface on the computer system side of the connector" (*id.*).

The Examiner states that since Bealkowski teaches enabling a communication interface to establish a communication link, as in claim 1, it teaches updating the state of the hot plug interface to indicate when the resources are ready to join the computing device. The Examiner states that by selectively enabling cards 11a-11d, the system must have knowledge of when the resources (the cashes of the cards) are ready to join. The Examiner also states that the LEDs indicate the status and green indicates that a processor card is ready to join (Ans. 14).

Appellants again argue that Bealkowski merely indicates that a service processor manages FET switches that isolate a processor card during addition and removal and there is "no teaching of the processor card updating a hot plug interface on the computer side of the connector" (Reply Br. 10).

*Issue*

Does Bealkowski teach "a midplane comprising a coupler and a hot plug interface to track a state associated with the coupler; . . . the hot plug module to update the state of the hot plug interface of the midplane to indicate when the resources are ready to join the computing device; and a processor coupled to the hot plug interface of the midplane, the processor to add the resources to the computing device without rebooting in response to

determining that the hot plug interface of the midplane indicates the resources are ready to join"?

*Analysis*

It would have been helpful if the Examiner had specifically identified the structures corresponding to the claim limitations at issue. Nevertheless, we find that Bealkowski anticipates claim 28 as broadly drafted.

We read the "hot plug interface" to correspond to the service processor 31 and hot plug controller 70 in Figure 2 of Bealkowski. The service processor and hot plug controller are part of the computing device to which the processor card 11a is coupled, which is considered the "midplane." Appellants do not contest that the computing device is a "midplane." The "coupler" is CPU connector 14 in Figure 3A. The hot plug controller 70 enables power and clock signals to the card 11a via FET switches 82, 86 (col. 6, l. 32 to col. 7, l. 12; col. 7, l. 51 to col. 8, l. 13). The service processor sends commands to the hot plug controller to execute initialization steps on the processor card 11a to prepare the card to be placed on the front side bus, such as power on configuration features and processor BIST (built-in self test) (col. 7, ll. 15-17; col. 8, ll. 14-39). The steps of the flowchart in Figures 4A and 4B represent states associated with the coupler which are tracked by the service processor and hot plug controller. The state is updated as it progresses through the steps. The processor card 11a is ready to join the computer device when a BIPI (bootstrap inter-processor interrupt) is received (col. 8, l. 66 to col. 9, l. 9; step 126, Fig. 4B). The

processor card 11a is added without rebooting in response to determining that the processor card 11a is ready to join, and the processor card 11a is joined when the FET switch 80a is enabled (col. 9, ll. 10-32). The LEDs also indicate when the module is ready to join.

Bealkowski teaches "a midplane comprising a coupler and a hot plug interface to track a state associated with the coupler; . . . the hot plug module to update the state of the hot plug interface of the midplane to indicate when the resources are ready to join the computing device; and a processor coupled to the hot plug interface of the midplane, the processor to add the resources to the computing device without rebooting in response to determining that the hot plug interface of the midplane indicates the resources are ready to join." The anticipation rejection of claim 28 is affirmed.

*Claim 29*

*Contentions*

Appellants argue that they are unable to locate the limitation, the "hot plug interface of the midplane to generate a hot plug interrupt in response to a change in the signal that is indicative of whether the coupler of the hot plug module has been coupled to the coupler of the midplane," in Bealkowski (Br. 15).

The Examiner refers to column 8, lines 1-5, which describes sending a CFG signal to detect the presence of a particular processor subsystem and

determining whether the presence of the particular processor subsystem is detected (Ans. 8).

Appellants argue that one skilled in the art would not consider a CFG signal similar to the claimed hot plug interrupt (Reply Br. 10-11).

*Issue*

Does Bealkowski teach "the hot plug interface of the midplane to generate a hot plug interrupt in response to a change in the signal that is indicative of whether the coupler of the hot plug module has been coupled to the coupler of the midplane"?

*Analysis*

The system determines whether the presence of the particular processor subsystem is detected in response to the CFG signal (col. 8, ll. 4-6). It seems that a signal indicating whether a particular subsystem has been detected is an "interrupt in response to a change in the signal that is indicative of whether the coupler of the hot plug module has been coupled to the coupler of the midplane." This signal, or interrupt, from step 108 is applied in the next step of the process in Figure 4A. Appellants do not state why the determination signal does not satisfy claim 29. In addition, it seems that the signal indicative of processor subsystem addition in step 102 of Figure 4A, which may be from hardware (col. 7, ll. 56-63), is indicative of whether a module has been coupled.

Bealkowski teaches "the hot plug interface of the midplane to generate a hot plug interrupt in response to a change in the signal that is

indicative of whether the coupler of the hot plug module has been coupled to the coupler of the midplane." The rejection of claim 29 is affirmed.

## CONCLUSION

The rejection of claims 1 and 7-9 under 35 U.S.C. § 102(e) over Olarig is affirmed.

The rejection of claims 1-5, 8-10, 28, and 29 under § 102(e) over Bealkowski is affirmed.

The rejection of claims 7, 35, and 37 under § 102(e) over Bealkowski is reversed.

Requests for extensions of time are governed by 37 C.F.R. § 1.136(b). *See* 37 C.F.R. § 41.50(f).

AFFIRMED-IN-PART

rwk

INTEL/BSTZ
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP
1279 OAKMEAD PARKWAY
SUNNYVALE, CA 94085-4040